



Windows Installation Guide for Suricata IDS/IPS/NSM

**This is a Suricata Windows Installation Guide -
Compilation from scratch.**

**Tested on Windows 7, Windows 8, Windows Server
2008R2, Server 2012 64 bit.**

Date: 15 Jan 2016

Document Version: 1.4.2

Author: Peter Manev(pevma)





INSTALLATION.....	4
CYGWIN - OVERVIEW OF INSTALLATION.....	4
CYGWIN - DOWNLOAD AND INSTALL.....	6
INSTALL SURICATA SPECIFIC DEPENDENCIES.....	15
SYSTEM VARIABLES - ADD PATHS.....	21
GET LIBPCAP - FOR WINDOWS.....	24
START CYGWIN.....	24
COMPILE SURICATA.....	25
Suricata from git - latest version.....	25
Suricata Stable, Beta or RC compilation.....	29
SET UP SURICATA FOR WINDOWS.....	32
Set up and copy needed config and dll files.....	32
Download rules.....	33
Adjust suricata.yaml configuration.....	34
CHECK ENABLED FEATURES FOR SURICATA.....	36
RUN SURICATA.....	38
Run Suricata on an un-ip'd interfaces.....	40





INFO AND DOCUMENTATION.....41





This is a guide of how to compile and come up with your own executable/binary of Suricata IDS/IPS on Windows. If you do not want to do that – there is a auto installation (MSI) windows native package here:

<http://suricata-ids.org/download/>

just run it and it will install and set up Suricata for you on your Windows system.

Installation

Cygwin - overview of installation

NOTE: Download - [setup-x86.exe](#) (32-bit installation). A compilation under 64 bit Cygwin installation will not work since [WinPcap](#) has only 32 bit downloads available and the compilation will fail.

After the installation is done you would need to add the packages below to your Cygwin installation - needed for Suricata to run:

libmpfr4, libmpfr-devel, mpfr, mingw-pthreads, gcc-core ,make, automake, automake1.9, zlib, zlib-devel, zlib0, autoconf, autoconf2.5, libtool , libglib2.0-devel, libglib2.0_0 ,pkg-config, libyaml-devel, libyaml0_2, libpcre1, libpcre-devel, file-devel, gcc-g++, wget

Extra and useful libraries/packages for enabling extra features during compile/make time or for compiling from git (latest devel version of Suricata):





luajit, luaji-devel, libGeoIP-devel, libGeoIP1, libnss-devel, libnss3, libnspr-devel, libnspr4, git



The above packages will allow us to enable during compile and build time the following extra features of Suricata -

- Lua (lua scripting)
- GeoIP
- MD5
- possibility to git clone the latest code if needed





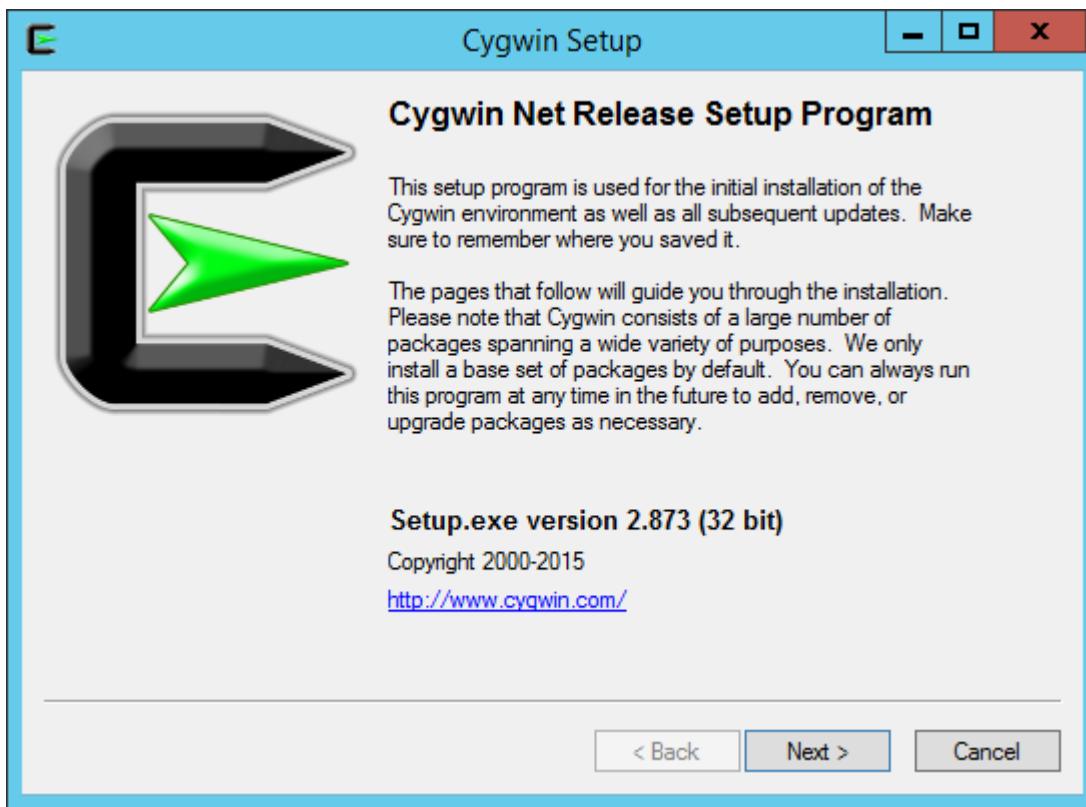
Cygwin - download and install

The following installations instructions were executed on Windows Server 2012R2 64 bit.

About 500 -600MB of space needed in total with all the necessary prerequisites installed.

Download <http://cygwin.com/setup-x86.exe> then double click the setup.exe to install

Go ahead and install it with the default options (basically just click next and ok)





Cygwin Setup - Choose Installation Type

Choose A Download Source
Choose whether to install or download from the internet, or install from files in a local directory.

Install from Internet
(downloaded files will be kept for future re-use)

Download Without Installing

Install from Local Directory

[< Back](#) [Next >](#) [Cancel](#)





Cygwin Setup - Choose Installation Directory

Select Root Install Directory
Select the directory where you want to install Cygwin. Also choose a few installation parameters.

Root Directory

Install For

All Users (RECOMMENDED)
Cygwin will be available to all users of the system.

Just Me
Cygwin will still be available to all users, but Desktop Icons, Cygwin Menu Entries, and important Installer information are only available to the current user. Only select this if you lack Administrator privileges or if you have specific needs.





Cygwin Setup - Select Local Package Directory

Select Local Package Directory

Select a directory where you want Setup to store the installation files it downloads. The directory will be created if it does not already exist.

Local Package Directory

< Back Cancel





Cygwin Setup - Select Connection Type

Select Your Internet Connection
Setup needs to know how you want it to connect to the internet. Choose the appropriate settings below.

Direct Connection
 Use Internet Explorer Proxy Settings
 Use HTTP/FTP Proxy:
Proxy Host
Port

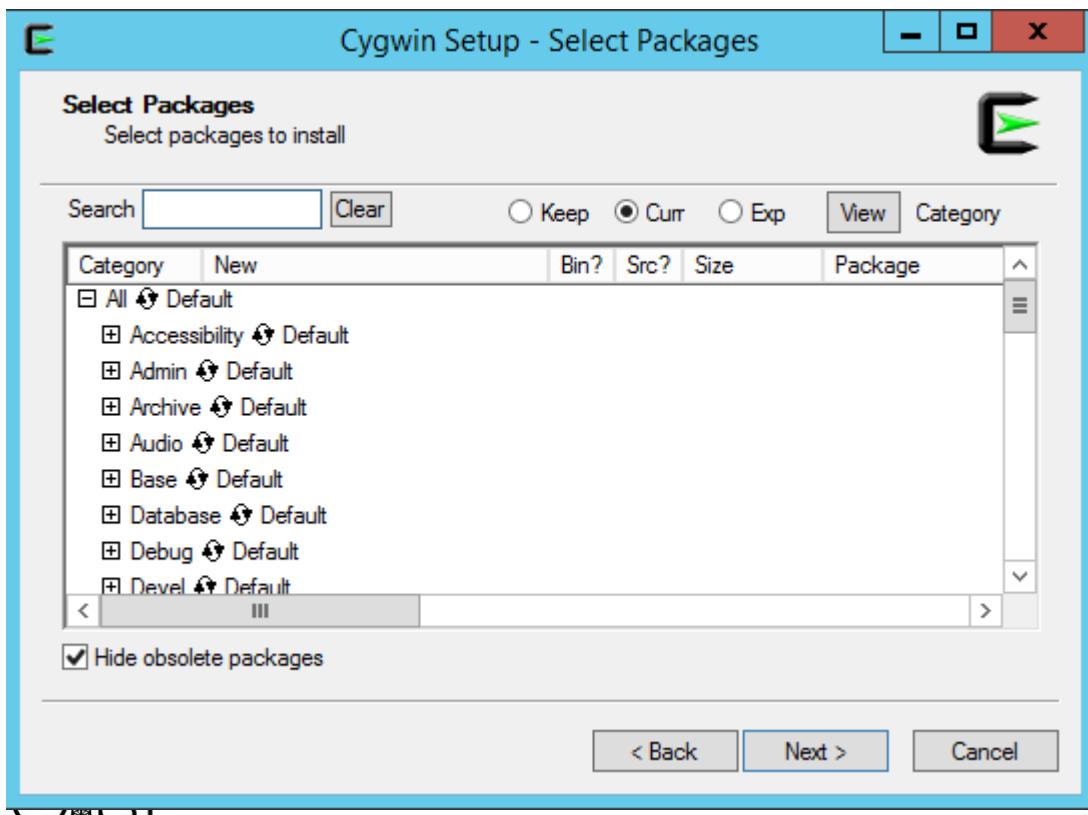
< Back Next > Cancel

Here , select any mirror you want:



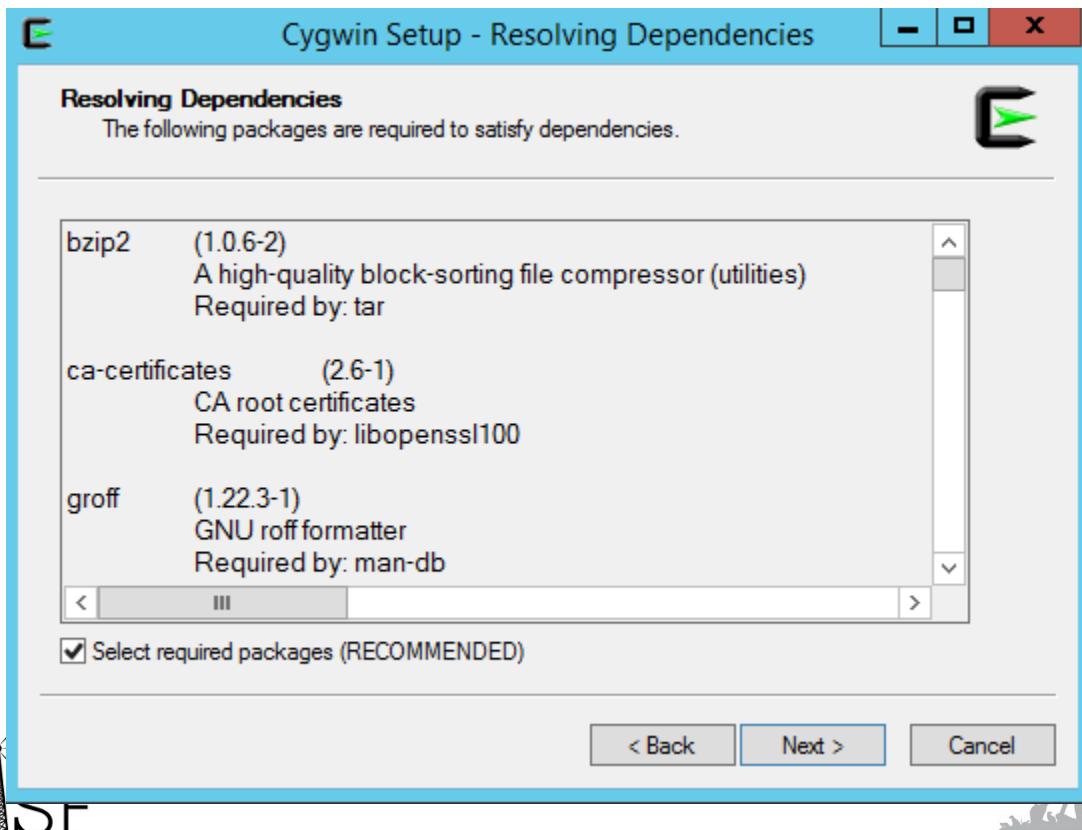


Click next to continue:



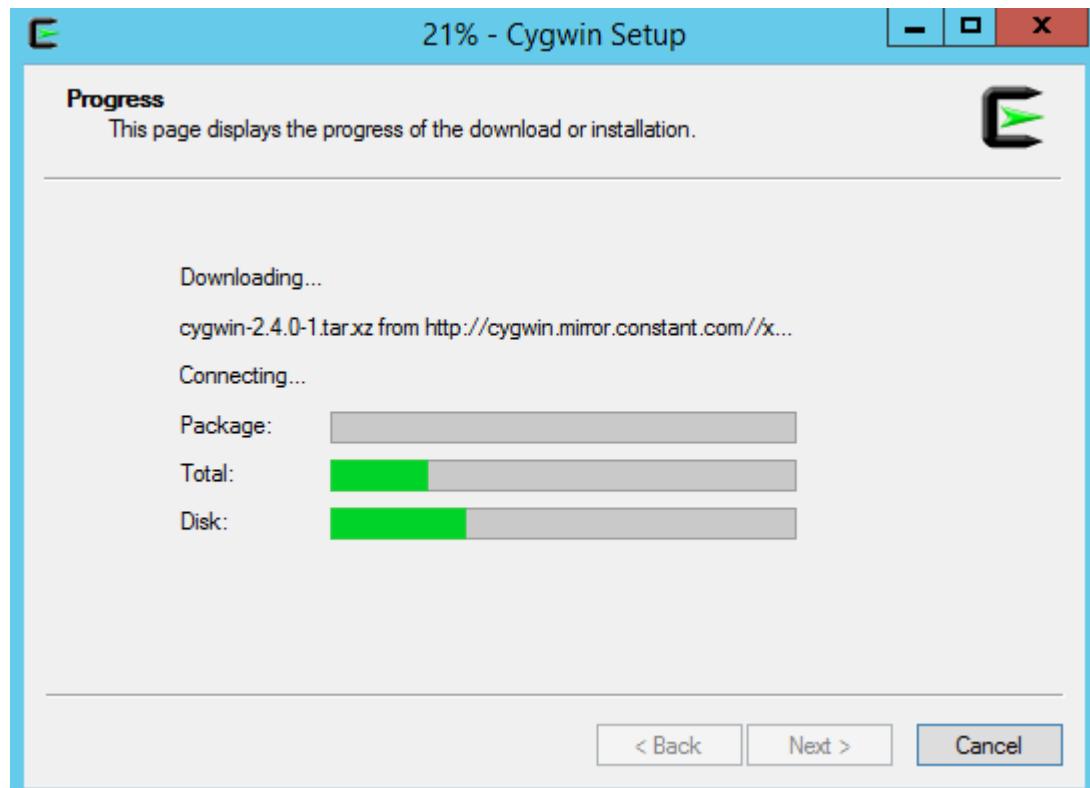


Next again to proceed with installation of the base packages:





Then you are going to see a progress bar:





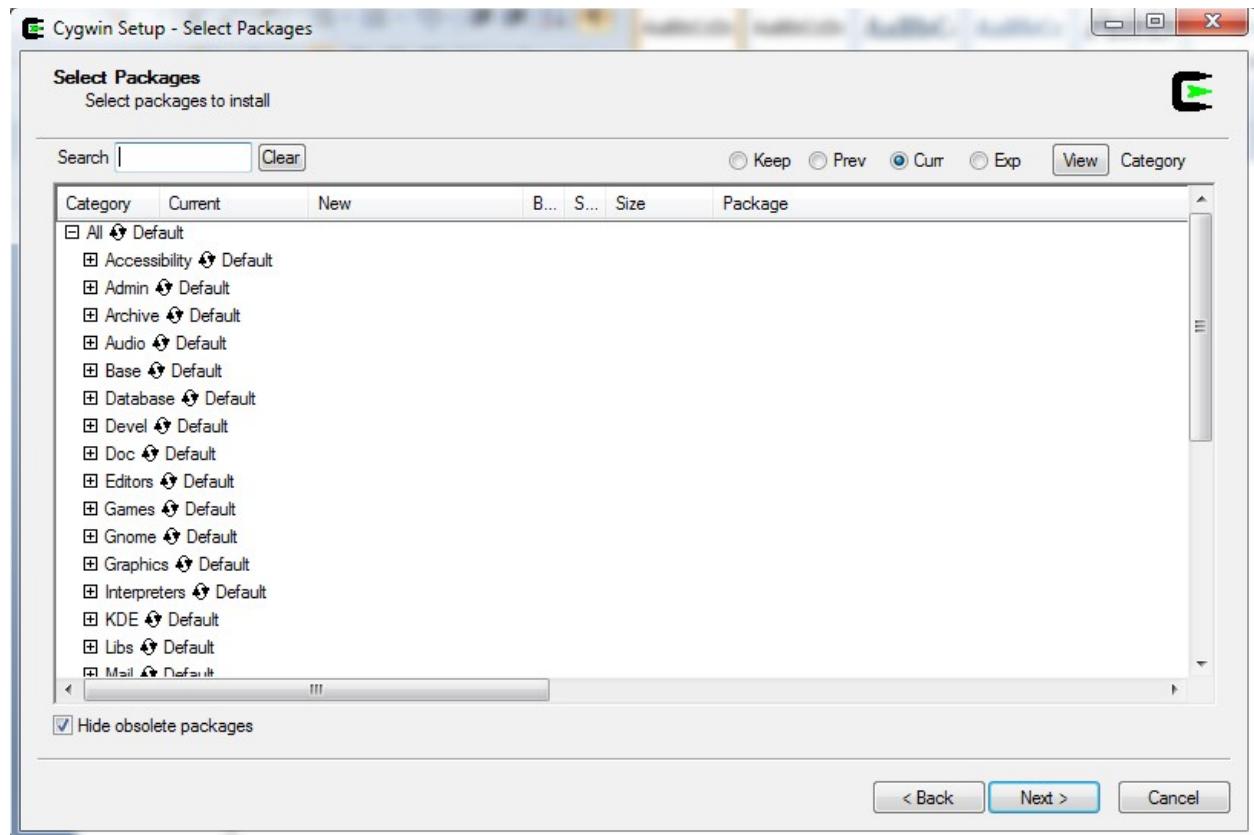


Install Suricata specific dependencies

After the installation is finished – we need to install the Suricata specific build dependencies (as described just before this section)

Go back and double-click the very same **setup-x86.exe** – we will need to install the extra packages necessary for Suricata to run.

Click next and ok until you are presented with the following screen:

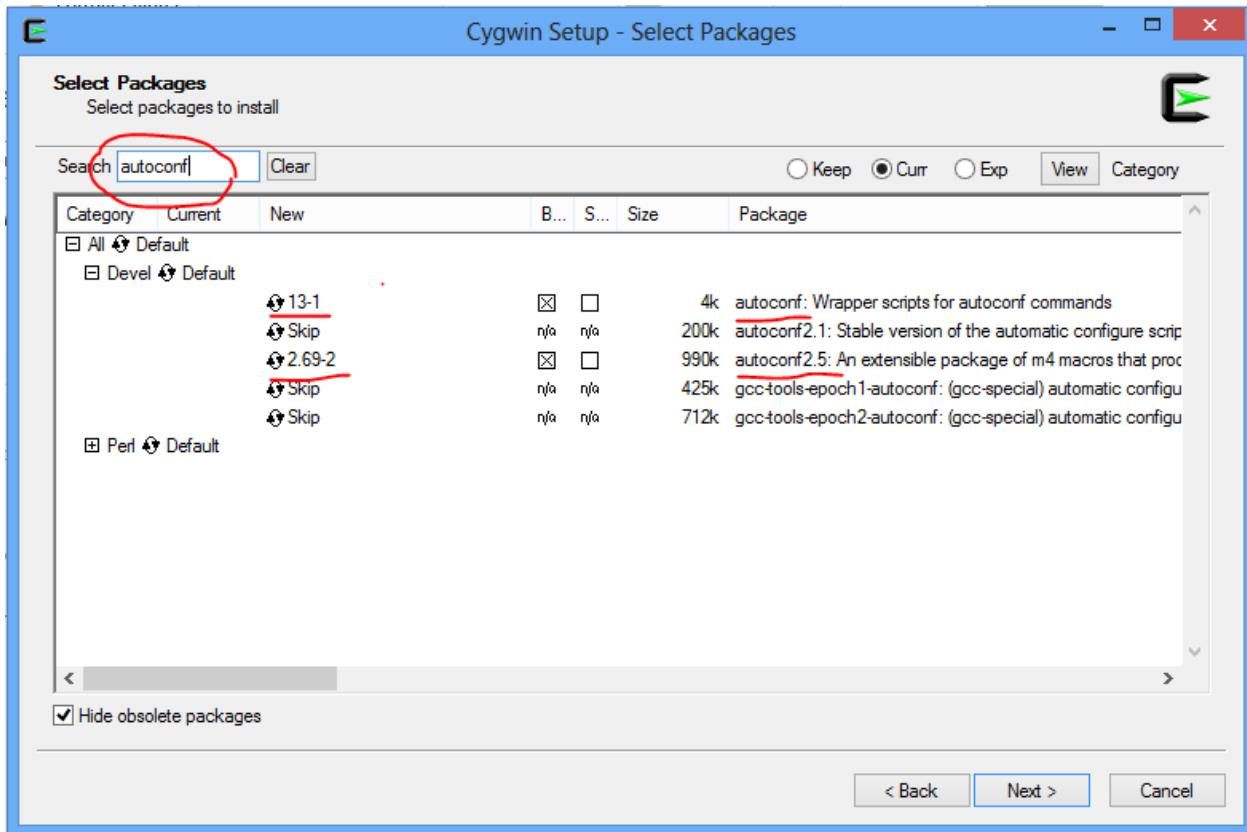




Here is where we search select and queue for installation the additional packages needed.

In the picture below , in the search box type in the name of the package- the search will return automatically , results , select the necessary package. Erase the contentment of the search box and type in the name of the next package, select ... and so on.

Do the same for all the needed packages, DO NOT hit next until you have selected all the packages.





The necessary packages are:

libmpfr4, libmpfr-devel, mpfr, mingw-pthreads, gcc-core ,make, automake, automake1.9, zlib, zlib-devel, zlib0, autoconf, autoconf2.5, libtool , libglib2.0-devel, libglib2.0_0 ,pkg-config, libyaml-devel, libyaml0_2, libpcre1, libpcre-devel, file-devel, gcc-g++, wget

And if you would like to enable extra functionality -

luajit, luaji-devel, libGeoIP-devel, libGeoIP1, libnss-devel, libnss3, libnspr-devel, libnspr4, git



Cygwin Setup - Select Packages

Select Packages
Select packages to install

Search dos2u Clear

Keep Curr Exp View Category

Category	Current	New	B...	S...	Size	Package
All Default	<input type="checkbox"/>					
Debug Default	<input type="checkbox"/>	Skip				50k dos2unix-debuginfo: Debug info for dos2unix
Text Default	<input type="checkbox"/>	6.0.4-1	<input checked="" type="checkbox"/>	<input type="checkbox"/>		67k dos2unix: Converts text files between DOS and *NIX li

< >

Hide obsolete packages

< Back Next > Cancel

This screenshot shows the Cygwin Setup - Select Packages window. The search bar at the top contains 'dos2u'. The 'Curr' radio button is selected under the filter options. The main table lists packages: 'dos2unix-debuginfo' (50k) and 'dos2unix' (67k). The 'dos2unix' package has its checkbox checked. A checkbox for 'Hide obsolete packages' is checked at the bottom. Navigation buttons for 'Back', 'Next >', and 'Cancel' are visible at the bottom right.





After you are done selecting the packages – make sure the “search” box is cleared, click the “view” button until the text on the right of the button displays “pending”.

Check and make sure all the needed packages are selected! If something is missing, go back and select it!

Click Next:

After that click next (make sure the option “select required packages (RECOMMENDED)” is selected!) :





Cygwin Setup - Resolving Dependencies

Resolving Dependencies

The following packages are required to satisfy dependencies.

autoconf2.1	(2.13-12)
Stable version of the automatic configure script builder	
Required by: autoconf, automake1.4, automake1.5	
automake1.10	(1.10.3-2)
(1.10) a tool for generating GNU-compliant Makefiles	
Required by: automake	
automake1.11	(1.11.6-2)
(1.11) a tool for generating GNU-compliant Makefiles	
Required by: automake, libtool	

Select required packages (RECOMMENDED)

< Back Next > Cancel

The extra packages that you have selected will start to download and install:





SURICATA™



0% - Cygwin Setup

Progress
This page displays the progress of the download or installation.

Downloading...
autoconf2.1-2.13-12.tar.bz2 from http://mirrors.163.com/cygwi...
Connecting...

Package:

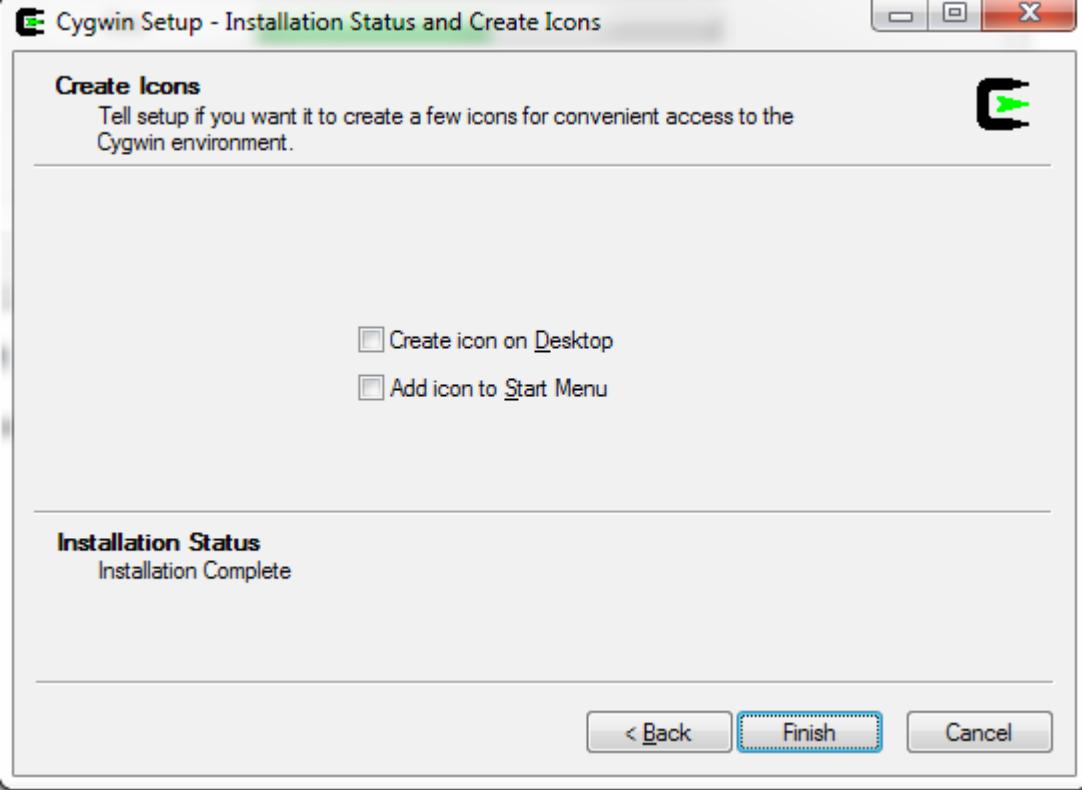
Total:

Disk:

< Back | Next > | Cancel

This could also take 5 min or so. Then click finish:





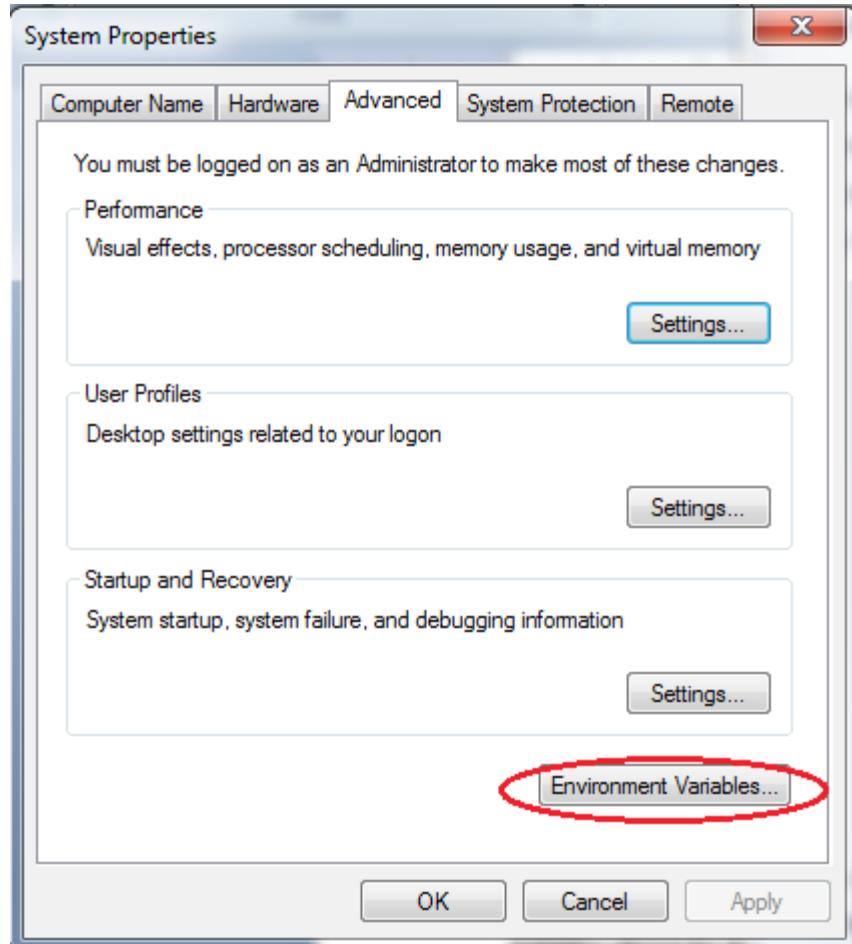
System variables - add paths

Add path to system variables (Win 7, Win 8, 2008, 2012 Server - Control Panel\System and Security\System\Advanced system settings\Environment Variables) :

C:\cygwin\bin;C:\cygwin\lib\pkgconfig;

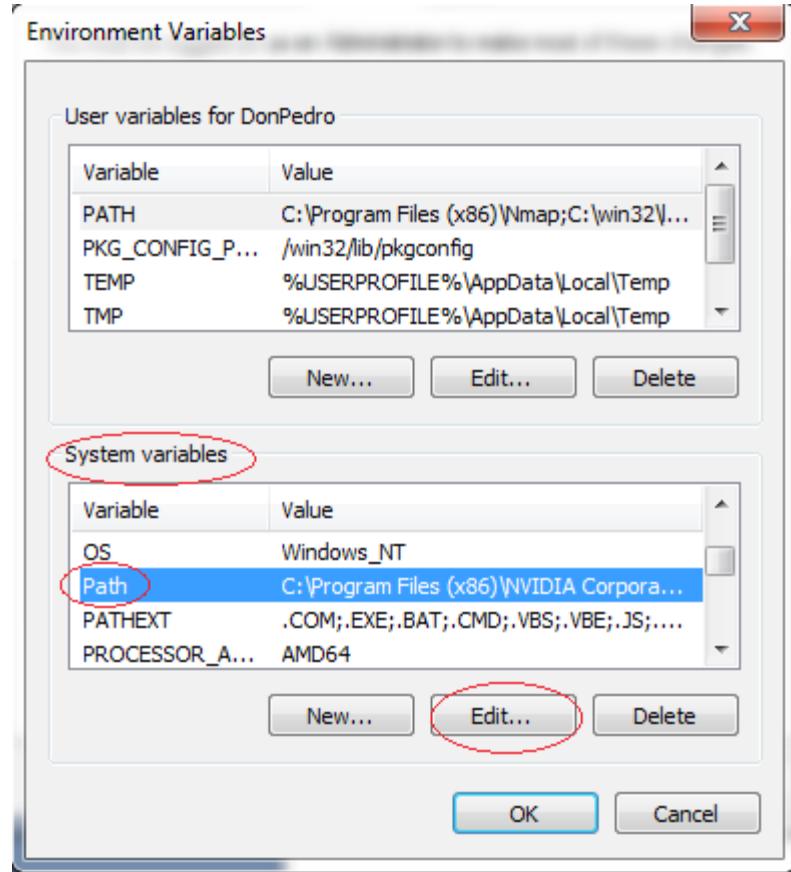
Add the above to environment system variables in your windows system!! See the picture below



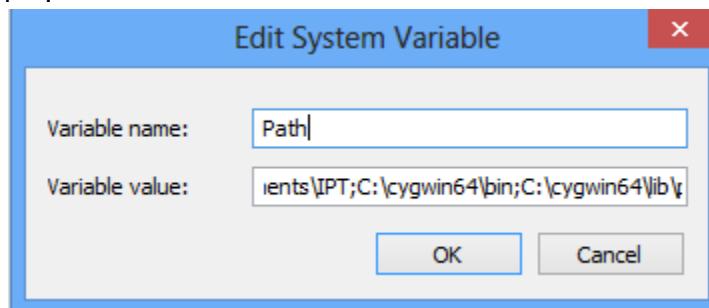


Edit the system path variable:





Add “ C:\cygwin\bin;C:\cygwin\lib\pkgconfig; “ without the quotes to the end of the “ Variable value path “ :





Get libpcap - for windows

Go to <http://www.winpcap.org/install/default.htm> and download the WinPcap installer for windows (at the time of this writing the current version was 4.1.3)

Install the WinPcap (double click, and just use the default options, basically click next and ok until finished.)

This is IMPORTANT , this is the development pack, we need that for Suricata to be able to run on Windows.

After that is done go to <http://www.winpcap.org-devel.htm>. Download the package and unpack it anywhere you like.

Copy libraries (from the unpacked directory) like this:

- ✓ **Copy ALL the content of WpdPack\Lib\ to cygwin\lib**
- ✓ **Rename “libwpcap” to “libpcap” (in your cygwin\lib\ directory)**
- ✓ **Copy all headers (all the content)from WpdPack\Include\ to C:\cygwin\usr\include**

Start Cygwin

Open CYGWIN. Double click your CYGWIN icon on your desktop. A Linux/bash like command prompt will open:





```
C ~ - X
Copying skeleton files.
These files are for the users to personalise their cygwin experience.

They will never be overwritten nor automatically updated.

'./.bashrc' -> '/home/Administrator//.bashrc'
'./.bash_profile' -> '/home/Administrator//.bash_profile'
'./.inputrc' -> '/home/Administrator//.inputrc'
'./.profile' -> '/home/Administrator//.profile'

Administrator@WIN-5B0EU82E444 ~
$
```

Compile Suricata

Suricata from git - latest version

(next section describes compilation for stable,beta,RC)

Get and compile Suricata.

As you are still in the CYGWIN environment -

Type in





git clone git://phalanx.openinfosecfoundation.org/oisf.git



Then after it is done

cd oisf

Then we need libhttp:

git clone git://github.com/ironbee/libhttp.git -b 0.5.x

```
Administrator@WIN-5B0EU82E444 ~
$ cd ../../
bin/           Cygwin.ico      etc/          proc/
cygdrive/       Cygwin-Terminal.ico   home/        sbin/
Cygwin.bat     dev/            lib/          tmp/          usr/
                                         var/

Administrator@WIN-5B0EU82E444 ~
$ cd ../../tmp/
Administrator@WIN-5B0EU82E444 /tmp
$ git clone git://phalanx.openinfosecfoundation.org/oisf.git
Cloning into 'oisf'...
remote: Counting objects: 42199, done.
remote: Compressing objects: 100% (11985/11985), done.
remote: Total 42199 (delta 34540), reused 36620 (delta 30167)
Receiving objects: 100% (42199/42199), 11.52 MiB | 316.00 KiB/s, done.
Resolving deltas: 100% (34540/34540), done.
Checking connectivity... done.

Administrator@WIN-5B0EU82E444 /tmp
$ cd oisf/
Administrator@WIN-5B0EU82E444 /tmp/oisf
$ git clone git://github.com/ironbee/libhttp.git -b 0.5.x
Cloning into 'libhttp'...
remote: Counting objects: 9963, done.
remote: Total 9963 (delta 0), reused 0 (delta 0), pack-reused 9963
Receiving objects: 100% (9963/9963), 9.92 MiB | 189.00 KiB/s, done.
Resolving deltas: 100% (6031/6031), done.
Checking connectivity... done.
Checking out files: 100% (1850/1850), done.

Administrator@WIN-5B0EU82E444 /tmp/oisf
$
```

Then we execute the following command(type and hit enter):





```
./autogen.sh && ./configure --enable-luajit --enable-pie --enable-geoip --disable-gccmarch-native --with-libnss-libraries=/usr/lib --with-libnss-includes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib --with-libnspr-includes=/usr/include/nspr/ && make clean && make
```

That will start configuration and compilation of Suricata.

The part -

```
--with-libnss-libraries=/usr/lib --with-libnss-includes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib --with-libnspr-includes=/usr/include/nspr/
```

will enable DM5s functionality for Suricata.

Like so:





A screenshot of a terminal window with a blue header bar. The title bar says '/tmp/oisf'. The window contains a black terminal session. At the top of the session, there is a single line of green text representing a command being typed or executed. The rest of the terminal window is blank black space.

```
Administrator@WIN-5B0EU82E444 /tmp/oisf
$ ./autogen.sh && ./configure --enable-luajit --enable-pie --enable-geoip --disable-gccmarch-native --with-libnss-libraries=/usr/lib --with-libnss-includes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib --with-libnspr-includes=/usr/include/nspr && make clean && make
```

Let it run....this could take a few minutes or so





After it is done your **suricata.exe** binary will be located under
src/.libs/suricata.exe:



```
Administrator@WIN-5B0EU82E444 /tmp/oisf
make[3]: Entering directory '/tmp/oisf/contrib'
make[3]: Nothing to be done for 'all-am'.
make[3]: Leaving directory '/tmp/oisf/contrib'
make[2]: Leaving directory '/tmp/oisf/contrib'
Making all in scripts
make[2]: Entering directory '/tmp/oisf/scripts'
Making all in suricatasc
make[3]: Entering directory '/tmp/oisf/scripts/suricatasc'
mkdir -p ../../scripts/suricatasc/src
./setup.py build;
running build
running build_py
creating build
creating build/lib
creating build/lib/suricatasc
copying src/suricatasc.py -> build/lib/suricatasc
copying src/_init_.py -> build/lib/suricatasc
running build_scripts
creating build/scripts-2.7
copying and adjusting suricatasc -> build/scripts-2.7
changing mode of build/scripts-2.7/suricatasc from 644 to 755
make[3]: Leaving directory '/tmp/oisf/scripts/suricatasc'
make[3]: Entering directory '/tmp/oisf/scripts'
make[3]: Nothing to be done for 'all-am'.
make[3]: Leaving directory '/tmp/oisf/scripts'
make[2]: Leaving directory '/tmp/oisf/scripts'
make[2]: Entering directory '/tmp/oisf'
make[2]: Leaving directory '/tmp/oisf'
make[1]: Leaving directory '/tmp/oisf'

Administrator@WIN-5B0EU82E444 /tmp/oisf
$ ls -lh src/.libs/
lt-suricata.c      suricata.exe          suricata_ltshwrapper

Administrator@WIN-5B0EU82E444 /tmp/oisf
$ ls -lh src/.libs/
total 22M
-rw-r--r-- 1 Administrator None  29K Jan 17 17:15 lt-suricata.c
-rwxr-xr-x 1 Administrator None 22M Jan 17 17:15 suricata.exe
-rw-r--r-- 1 Administrator None  6.2K Jan 17 17:15 suricata_ltshwrapper

Administrator@WIN-5B0EU82E444 /tmp/oisf
$ |
```

Suricata Stable, Beta or RC compilation

As you are still in the CYGWIN environment -

This section uses Suricata 3.0RC3 as an example.





If you want to install Suricata stable you can find it here - <http://suricata-ids.org/download/>



go to a tmp dir. Type in and hit enter to complete each step:

- 1) `wget http://www.openinfosecfoundation.org/download/suricata-3.0RC3.tar.gz`
- 2) `tar -zxf suricata-3.0RC3.tar.gz`
- 3) `cd suricata-3.0RC3`
- 4) `libtoolize -c && autoreconf -fv --install && ./configure --enable-luajit --enable-pie --enable-geoip --disable-gccmarch-native --with-libnss-libraries=/usr/lib --with-libnss-includes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib --with-libnspr-includes=/usr/include/nspr && make clean && make`

The part -

`-with-libnss-libraries=/usr/lib --with-libnss-includes=/usr/include/nss/ --with-libnspr-libraries=/usr/lib -with-libnspr-includes=/usr/include/nspr`

will enable DM5s functionality for Suricata.

NOTE: Please note the difference in the compilation line (4 above) for stable/beta/RC and for git.

After done the **suricata.exe** binary will be located in the folder
/src/.libs/suricata.exe





```
Administrator@WIN-5B0EU82E444 /tmp/suricata-3.0RC3
$ running build_scripts
creating build/scripts-2.7
copying and adjusting suricatasc -> build/scripts-2.7
changing mode of build/scripts-2.7/suricatasc from 644 to 755
make[3]: Leaving directory '/tmp/suricata-3.0RC3/scripts/suricatasc'
make[3]: Entering directory '/tmp/suricata-3.0RC3/scripts'
make[3]: Nothing to be done for 'all-am'.
make[3]: Leaving directory '/tmp/suricata-3.0RC3/scripts'
make[2]: Leaving directory '/tmp/suricata-3.0RC3/scripts'
make[2]: Entering directory '/tmp/suricata-3.0RC3'
make[2]: Leaving directory '/tmp/suricata-3.0RC3'
make[1]: Leaving directory '/tmp/suricata-3.0RC3'

Administrator@WIN-5B0EU82E444 /tmp/suricata-3.0RC3
$

Administrator@WIN-5B0EU82E444 /tmp/suricata-3.0RC3
$ ls -lh src/.libs/
total 22M
-rw-r--r-- 1 Administrator None 29K Jan 17 17:34 lt-suricata.c
-rwxr-xr-x 1 Administrator None 22M Jan 17 17:34 suricata.exe
-rw-r--r-- 1 Administrator None 6.2K Jan 17 17:34 suricata_ltshwrapper

Administrator@WIN-5B0EU82E444 /tmp/suricata-3.0RC3
$
```

Next steps.

For the instructions below if you want to use stable or RC3 (as opposed to latest git Suricata) – just substitute the **oisf** directory with the appropriate name – **suricata-3.0RC3** for example.





Set up Suricata for Windows



Set up and copy needed config and dll files

Create the following directories:

- C:\Program Files (x86)\Suricata\log
- C:\Program Files (x86)\Suricata\log\files
- C:\Program Files (x86)\Suricata\log\certs
- C:\Program Files (x86)\Suricata\rules

Then copy the suricata.exe file from C:\cygwin\tmp\oisf\src\libs to C:\Program Files (x86)\Suricata

NOTE: It is not a must to place Suricata in C:\Program Files (x86)\Suricata you can place it anywhere you would like.

Copy (from C:\cygwin\bin)

1. cyggcc_s-1.dll
2. cygGeolP-1.dll
3. cygluajit-5.1-2.dll
4. cygmagic-1.dll
5. cygnspr4.dll
6. cygnss3.dll
7. cygnssutil3.dll
8. cygpcre-1.dll
9. cygplc4.dll





10.cygplds4.dll

11.cygwin1.dll

12.cygz.dll



to your C:\Program Files (x86)\Suricata directory

Also copy C:\cygwin\usr\share\misc\magic.mgc to your C:\Program Files (x86)\Suricata directory

Download rules

Go to <http://rules.emergingthreats.net/open/suricata/>

Download a rule set.

<http://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz>

Unzip/untar the rule set in the C:\Suricata\rules directory.

Then go to C:\cygwin\tmp\oisf

Copy

classification.config , reference.config and suricata.yaml to

C:\Program Files (x86)\Suricata





Adjust suricata.yaml configuration

Open suricata.yaml with an editor - Notepad, Notepad++, whichever you like and change the following lines:

```
# The default logging directory. Any log or output file will be
# placed here if its not specified with a full path name. This can be
# overridden with the -l command line parameter.

default-log-dir: C:\\Program Files (x86)\\Suricata\\log\\

...
...

# Magic file. The extension .mgc is added to the value here.

#magic-file: /usr/share/file/magic

magic-file: C:\\Program Files (x86)\\Suricata\\magic.mgc

...
...

outputs:
  - console:
      enabled: yes
      # type: json
  - file:
      enabled: yes
```





filename: C:\|Program Files (x86)\|Suricata\|log\|suricata.log

type: json

...

...

Set the default rule path here to search for the files.

if not set, it will look at the current working dir

default-rule-path: C:\|Program Files (x86)\|Suricata\|rules\|

rule-files:

...

...

classification-file: C:\|Program Files (x86)\|Suricata\classification.config

reference-config-file: C:\|Program Files (x86)\|Suricata\reference.config

...

...

vars:

Holds the address group vars that would be passed in a Signature.

These would be retrieved during the Signature address parsing stage.

address-groups:

HOME_NET: "[192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]" (adjust network ranges here to the ones that you want Suricata to inspect)

EXTERNAL_NET: "!\$HOME_NET"





HTTP_SERVERS: "\$HOME_NET"

SMTP_SERVERS: "\$HOME_NET"



Check enabled features for Suricata

Open a cmd as ADMINISTRATOR!!!.

Got to C:\Program Files (x86)\Suricata and execute

suricata.exe -build-info





Administrator: Command Prompt

```
C:\>Program Files (<x86>)\Suricata>
C:\>Program Files (<x86>)\Suricata>suricata.exe --build-info
This is Suricata version 3.0dev (rev 44a444b)
Features: PCAP_SET_BUFF LIBPCAP_VERSION_MAJOR=1 HAVE_PACKET_FANOUT HAVE_HTP_URI_
NORMALIZE_HOOK PCRE_JIT HAVE_NSS HAVE_LUA HAVE_LUAJIT TLS
SIMD support: none
Atomic intrinsics: 1 2 4 8 byte(s)
32-bits, Little-endian architecture
GCC version 4.9.3, C version 199901
L1 cache line size <CLS>=64
thread local storage method: __thread
compiled with LibHTTP v0.5.18, linked against LibHTTP v0.5.18

Suricata Configuration:
  AF_PACKET support: no
  PF_RING support: no
  NFQueue support: no
  NFLOG support: no
  IPFW support: no
  Netmap support: no
  DAG enabled: no
  Napatech enabled: no

  Unix socket enabled: no
  Detection enabled: yes

  libnss support: yes
  libnspr support: yes
  libjansson support: no
  hiredis support: no
  Prelude support: no
  PCRE jit: yes
  LUA support: yes, through luajit
  libluajit: yes
  libgeoip: yes
  Non-bundled http: no
  Old barnyard2 support: no
  CUDA enabled: no

  Suricatasc install: yes

  Unit tests enabled: no
  Debug output enabled: no
  Debug validation enabled: no
  Profiling enabled: no
  Profiling locks enabled: no
  Coccinelle / spatch: no

Generic build parameters:
  Installation prefix: /usr/local
  Configuration directory: C:\Program Files (<x86>)\Suricata\
  Log directory: C:\Program Files (<x86>)\Suricata\log

  --prefix NONE
  --sysconfdir /usr/local/etc
  --localstatedir /usr/local/var

  Host: i686-pc-cygwin
  Compiler: gcc <exec name> / gcc <real>
  GCC Protect enabled: no
  GCC march native enabled: no
  GCC Profile enabled: no
  Position Independent Executable enabled: yes
  CFLAGS -g -O2
  PCAP_CFLAGS
  SECCFLAGS

C:\>Program Files (<x86>)\Suricata>
```





Run Suricata



Open a cmd as ADMINISTRATOR!!!.

Got to C:\Program Files (x86)\Suricata and execute

C:\Program Files (x86)\Suricata>suricata.exe -c suricata.yaml -i 10.0.2.15 -v

like shown on the picture below (in this case - 10.0.2.15 is the IP/interface I want Suricata to listen to, i.e. the IP that my network card has been configured with):

```
C:\> Administrator: Command Prompt - suricata.exe -c suricata.yaml -i 10.0.2.15 -v
C:\Program Files (x86)\Suricata>suricata.exe -c suricata.yaml -i 10.0.2.15 -v
[136] 17/1/2016 -- 20:08:34 - <suricata.c:1542> <Info> <ParseCommandLine> -- translated 10.0.2.15 to pcap device \Device\NPF_{156DACP3-585B-400A-AC12-AAACFE839870
[136] 17/1/2016 -- 20:08:34 - <suricata.c:1073> <Notice> <SCPrintVersion> -- This
is Suricata version 3.0dev (rev 44a444b)
[136] 17/1/2016 -- 20:08:34 - <util-cpu.c:170> <Info> <UtilCpuPrintSummary> -- CPUs/cores online: 1
[136] 17/1/2016 -- 20:08:34 - <app-layer-htp.c:2251> <Info> <HTTPConfigSetDefault
sPhase2> -- 'default' server has 'request-body-minimal-inspect-size' set to 3388
2 and 'request-body-inspect-window' set to 4053 after randomization.
[136] 17/1/2016 -- 20:08:34 - <app-layer-htp.c:2266> <Info> <HTTPConfigSetDefault
sPhase2> -- 'default' server has 'response-body-minimal-inspect-size' set to 421
19 and 'response-body-inspect-window' set to 16872 after randomization.
[136] 17/1/2016 -- 20:08:34 - <app-layer-dns-udp.c:337> <Info> <DNSUDPConfigure>
-- DNS request flood protection level: 500
[136] 17/1/2016 -- 20:08:34 - <app-layer-dns-udp.c:349> <Info> <DNSUDPConfigure>
-- DNS per flow memcap <state-memcap>: 524288
[136] 17/1/2016 -- 20:08:34 - <app-layer-dns-udp.c:361> <Info> <DNSUDPConfigure>
-- DNS global memcap: 16777216
```

And you have yourself Suricata running (the start time could depend the PC/Server CPU/MEM availability and of course how many rules and what options you have enabled in suricata.yaml):





SURICATA™



```
[136] 17/1/2016 -- 20:08:34 - <stream-tcp.c:475> <Info> <StreamTcpInitConfig> -- stream.reassembly "memcap": 134217728
[136] 17/1/2016 -- 20:08:34 - <stream-tcp.c:493> <Info> <StreamTcpInitConfig> -- stream.reassembly "depth": 1048576
[136] 17/1/2016 -- 20:08:34 - <stream-tcp.c:576> <Info> <StreamTcpInitConfig> -- stream.reassembly "tosever-chunk-size": 2537
[136] 17/1/2016 -- 20:08:34 - <stream-tcp.c:528> <Info> <StreamTcpInitConfig> -- stream.reassembly "toclient-chunk-size": 2600
[136] 17/1/2016 -- 20:08:34 - <stream-tcp.c:591> <Info> <StreamTcpInitConfig> -- stream.reassembly.raw: enabled
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 4, prealloc 256
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 16, prealloc 512
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 112, prealloc 512
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 248, prealloc 512
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 512, prealloc 512
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 768, prealloc 1024
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 1448, prealloc 1024
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:451> <Info> <StreamTcpReassembleConfig> -- segment pool: pktsize 65535, prealloc 128
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:487> <Info> <StreamTcpReassembleConfig> -- stream.reassembly "chunk-prealloc": 250
[136] 17/1/2016 -- 20:08:34 - <stream-tcp-reassemble.c:500> <Info> <StreamTcpReassembleConfig> -- stream.reassembly "zero-copy-size": 128
[136] 17/1/2016 -- 20:08:34 - <ippair.c:211> <Info> <IPPairInitConfig> -- allocated 262144 bytes of memory for the ippair hash... 4096 buckets of size 64
[136] 17/1/2016 -- 20:08:34 - <ippair.c:234> <Info> <IPPairInitConfig> -- preallocated 1000 ippairs of size 72
[136] 17/1/2016 -- 20:08:34 - <ippair.c:236> <Info> <IPPairInitConfig> -- ippair memory usage: 334144 bytes, maximum: 16777216
[136] 17/1/2016 -- 20:08:34 - <util-magic.c:62> <Info> <MagicInit> -- using magic file C:\Program Files (x86)\Suricata\magic.mgc
[136] 17/1/2016 -- 20:08:34 - <suricata.c:1950> <Info> <SetupDelayedDetect> -- Delayed detect disabled
[136] 17/1/2016 -- 20:08:34 - <reputation.c:620> <Info> <SRepInit> -- IP reputation disabled
[136] 17/1/2016 -- 20:08:34 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\botcc.rules
[136] 17/1/2016 -- 20:08:34 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\ciarmy.rules
[136] 17/1/2016 -- 20:08:34 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\compromised.rules
[136] 17/1/2016 -- 20:08:34 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\drop.rules
[136] 17/1/2016 -- 20:08:34 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\dshield.rules
[136] 17/1/2016 -- 20:08:34 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-activex.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-attack_response.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-chat.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-current_events.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-dns.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-dos.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading rule file: C:\Program Files (x86)\Suricata\rules\emerging-exploit.rules
[136] 17/1/2016 -- 20:08:35 - <detect.c:416> <Info> <ProcessSigFiles> -- Loading
```





Run Suricata on an un-ip'd interfaces



If you need to run Suricata on an un-ip'd interfaces(thanks to Rich Rumble for pointing that out):

You can get the NIC UUID in a variety of ways, the simplest is using a single command for WMIC:(from cmd prompt paste in the following)

```
wmic nicconfig get ipaddress,SettingID
```

If you know your NIC's IP you can filter the results with findstr:

```
wmic nicconfig get ipaddress,SettingID | findstr 1.2.3.4
```

(replace 1.2.3.4 with your NIC's IP)

Then use that as your interface argument:

```
suricata.exe -c suricata.yaml -i \\DEVICE\\NPF_{EE7B2A76-9343-449F-B3D8-3CB0F37DCA49}
```

Make sure the double slashes are used, and a backslash is placed before the curly braces!

That's it.

From here on it is up to you to configure Suricata the way it suits you best!

Thanks for trying Suricata!





Info and documentation



You can find much more info about setting up and tuning Suricata here:

<https://redmine.openinfosecfoundation.org/projects/suricata/wiki>

